In the claims:

For the Examiner's convenience, all pending claims are presented below with changes shown in accordance with the new format.

1. (Previously Presented) For a computer-executable program that operates on a data structure, where the data structure must have a required state at selected program points, a method of transforming said program comprising the steps of:

(A) analyzing the program to determine the state of said data structure at said selected program points;

(B) partitioning said determined state at each said program point into components that may each be set separately;

(C) determining the operations required to set each component of the state at each selected program point; and

(D) placing said operations to eliminates partial redundancies of said operations.

2. (Original) The method of claim 1, wherein the data structure stores items on a first-in-last-out basis.

3. (Original) The method of claim 2, wherein the states of the data structure are represented as paths on a tree of nodes where:

(A) each path traverses the tree towards the root; and

(B) each node on the path represent a component of the state.

4. (Previously Presented) The method of claim 2, wherein the data structure represents actions to be taken by the program if an exception occurs.

5. (Previously Presented) The method of claim 4, wherein the selected program points are the points of execution immediately before instructions that might cause an exception.

6. (Previously Presented) The method of claim 4, further comprising representing the actions to be taken as exception paths in a graph.

7-9. (Canceled)

~~7~~ 10. (Currently Amended) For a computer-executable program that operates on a data structure, where the data structure must have a required state at selected program points, a method of transforming said program comprising the steps of:

(A) analyzing the program to determine the state of an instance of said data structure at said selected program points;

(B) partitioning said instance of said data structure into components;

(C) determining a set of one or more operations for setting the components;

(D) computing placement of the set of operations to eliminate partial redundancies; and

(E) inserting the set of operations at said program points according to the computed placement.

~~8~~ 11. (Currently Amended) The method of ~~claim 7~~ claim 10 wherein the data structure is an exception handling stack.

~~9~~ 12. (Currently Amended) The method of ~~claim 8~~ claim 11 wherein the components are a pointer to the exception handling stack and an exception handling data structure.

~~10~~ 13. (Currently Amended) A machine-readable medium having a set of instructions, which when executed by a set of one or more processors, causes said set of processors to perform operations comprising:

(A) analyzing a program that operates on a data structure, which must have a required state at selected program points in the program, to determine the state of an instance of said data structure at said selected program points;

(B) partitioning said instance of said data structure into components;

(C) determining a set of one or more operations for setting the components;

(D) computing placement of the set of operations to eliminate partial redundancies; and

(E) inserting the set of operations at said program points according to the computed placement.

~~11~~ 14. (Currently Amended) The machine-readable medium of ~~claim 10~~ claim 13, wherein the data structure stores items on a first-in-last-out basis.

~~12~~ 15. (Currently Amended) The machine-readable medium of ~~claim 11~~ claim 14, wherein the states of the data structure are represented as paths on a tree of nodes where:

(A) each path traverses the tree towards the root; and

(B) each node on the path represent a component of the state.

~~13~~ 16. (Currently Amended) The machine-readable medium of ~~claim 11~~ claim 14, wherein the data structure represents actions to be taken by the program if an exception occurs.

~~14~~ 17. (Currently Amended) The machine-readable medium of ~~claim 13~~ claim 16, wherein the selected program points are the points of execution immediately before instructions that might cause an exception.

C

~~15~~ 18. (Canceled)